

Día de la Aceptación Universal Uruguay - 26 de mayo de 2025

Sesión Teórica: Conceptos clave de Aceptación Universal

Nicolás Antoniello – ICANN

Carlos Martínez – LACNIC

Sesión Teórica: Conceptos clave de Aceptación Universal

Nicolás Antoniello – ICANN
Carlos Martínez – LACNIC

Carlos Martínez: Bueno, nosotros venimos a guardar la fiesta un poco, porque escuchamos durante la mañana una serie de exposiciones que comentan todo lo bueno y lo deseable que es tener un internet multilingüe, dominios en múltiples caracteres, etc. O porque alguien lo tiene que hacer eso, ¿no? Entonces nosotros

venimos a mostrar, les diría que rascar apenas la complejidad que se esconde detrás de eso.

Yo, y me reconozco históricamente un escéptico del tema, a mí sí me parecía que se exageraba el problema hasta que me puse a estudiarlo un poco, y es la complejidad que tiene esto es increíble y nosotros la minimizamos porque en realidad nosotros tenemos, en el idioma que hablamos, cinco, seis y quizás alguno de los más viejos se acuerde del séptimo carácter especial, que son los cinco acentos, la ñ y el diéresis. Eso es lo único que tenemos. Y el diéresis, yo creo que mi hijo no sabe lo que es, por ejemplo.

Otros sets de caracteres tienen complejidades mucho, mucho, mucho, mucho peores, como por ejemplo hasta sets de caracteres donde los caracteres se diferencian en su versión angosta o versión ancha. La misma letra, pero es ancha y angosta y significa distintas cosas. Cada una de esas cosas se tiene que codificar de alguna manera.

Entonces, bueno, nosotros hoy, entre esta parte de la mañana y lo de la tarde, les vamos a mostrar... Contar un poco de esa complejidad y mostrarles cómo encarar un poco y hacer algunas cosas con un servidor de correo electrónico y con un dominio. Nico, vos que supuestamente sos el que sabe de todo esto. Adelante.

Nicolás Antoniello: Bueno, he estado tomando un poco de Coca-Cola porque me estaban diciendo por ahí mis compañeros de ICANN, no comas mi alfajor de maicena antes de hablar, digamos, vas a empezar a... Está complicado, no coman alfajor de maicena antes de hablar. Moraleja de hoy.

Bien. Como dijo Cristian hoy más temprano, pregunten, interrumpan, todo lo que quieran. De nuevo, este tipo de presentaciones, se supone que nosotros ya lo tenemos que saber, lo que estamos presentando. Así que son más para ustedes, para que saquen las dudas. Y además con Carlos nos gusta hacer estas presentaciones que tienen un... A propósito, le dejamos un gran porcentaje de improvisación. Porque creo que las hace más dinámica. Pero gran parte de eso también reside en que ustedes hagan preguntas e interactuemos.

La presentación en sí va a ser una guía, capaz que no llegamos al final de la presentación. Y va a depender un poco de cómo se vaya dando la conversación. Sí se va a poner un poquito más técnico, un poquito, no más. Y en la tarde se va a poner un poquito más técnico que ahora. Pero la idea también es que lo puedan... Ir siguiendo todos, aunque no sean técnicos. ¿Sí?

Bien. Esto ya lo dijeron hoy más temprano, varias veces. Así que va a ser tipo repaso rápido. ¿Qué es la aceptación universal? Básicamente, soportar escritura en todos los sistemas, aplicaciones, en otros lenguajes escritos que no sean código ASCII. O sea, básicamente que no sea inglés. ¿Sí? Y el código ASCII ni siquiera es todo el inglés, digamos, porque hay un montón de caracteres en inglés, separadores, puntuaciones, etcétera, que ni siquiera están contempladas en el código ASCII. Esto no es inglés. Esto es código ASCII, que es todavía más restrictivo que el inglés.

Esa es la versión original del DNS, el sistema de nombres de dominio. Solo soportaba caracteres ASCII, codificación ASCII en los nombres de dominio. Y correo electrónico también, solo caracteres ASCII para los nombres de las casillas de correo. Y bueno, obviamente, del lado del DNS, lo que está antes del arroba, que es el nombre de la casilla, y lo que está después del arroba. O sea, a la izquierda del arroba el nombre de la casilla, a la derecha del arroba el nombre de dominio. Ninguno de los dos, en la versión original de esos estándares, se podía escribir en otra cosa que no fuera caracteres ASCII. ¿Sí?

Y voy a hacer un spoiler. Voy a adelantar algo que está al final de la presentación. Porque no es solo correo electrónico. Correo electrónico es una parte de lo que es aceptación universal. De hecho, esta parte de la aceptación universal, van a ver por ahí que aparece una sigla en inglés, que creo que es algo así como Internationalized Email Address, IEA. El IEAS es la versión de correo electrónico de la aceptación universal, digamos.

Pero, de nuevo, de nada sirve tener todos los servidores de correo con aceptación. Y cuando voy a pedirle para acceder a mi sitio web a un usuario que cree, se cree un usuario y elija un nombre de usuario y una contraseña, después le digo que elija, pero en realidad es mentira, no lo dejo elegir mucho porque solamente puede usar su correo electrónico con el nombre de usuario y la contraseña sí la puede elegir, por suerte. Pero cuando va a escribir el nombre de correo, la dirección de correo electrónico, si mi correo electrónico fuera nicolás arroba, no sé, internet.com.uy, creo que ya no existen más esos. De hecho, el mío era nicolás arroba internet.com.uy sin tilde, porque no podía poner tilde. Ahora pongo el tilde, pero cuando voy a la página web, no me lo acepta, porque el parser, la pieza de software, digamos, que fue utilizada para ver si lo que estoy poniendo ahí es un correo electrónico, cuando ve algo que no es ASCII, dice, ¿cómo que no es un correo electrónico? ¿Cómo que no es un correo electrónico? Está un poquito desactualizado.

Se trata de actualizar entonces también todos los sitios web y en general todas las aplicaciones para que tengan soporte de múltiples escrituras, digamos. Bueno, esa es la aceptación universal.

¿Por qué importa? También ya lo dijimos, apoyar un internet diversa y multilingüe, permitir mayor competencia, innovación y elección del consumidor, crear oportunidades de negocio, eso. Desde mi punto de vista personal, es bastante relativo, pero también es importante y es uno de los impulsores de todo en el mundo, ¿no? Y, bueno, ayudar a los gobiernos a desarrollar políticas y también, no está dicho explícitamente acá, pero está implícito, digamos, hay mucho de mantener la identidad, preservar la identidad cultural, ¿no?

Hay personas que no pueden escribir en otro lenguaje. Hay personas que no pueden escribir en otro lenguaje que no sea el propio y no conocen otros lenguajes. Y hay otras que con todo el derecho del mundo sí saben tres o cuatro o cinco idiomas, muchísimos idiomas más de lo que yo puedo saber, pero no quieren escribir en otro idioma porque quieren preservar su identidad. Entonces, esto también se trata de eso, ¿no?

Bien. Entonces, vamos a hablar un poco de lo que es, un poco más en detalle, lo que es aceptación universal desde el punto de vista de los nombres de dominio, o sea, desde el punto de vista del sistema de nombres de dominio, y lo que es aceptación universal desde el punto de vista de las direcciones de correo electrónico, ¿sí?

Levante la mano el que no sabe lo que es el sistema de nombres de dominio. No tengan, esto no es una prueba, así que, y no tienen por qué saber todo lo que es el sistema de nombres de dominio. Todos saben lo que es. Por lo menos todos creen que saben lo que es. Y después cuando uno le pregunta, dice, yo sé lo que es, pero no sé explicarlo. Eso pasa, nos pasa a todos. Porque saber explicarlo no es lo mismo que saber qué es algo, digamos. ¿Sí? Bueno, se supone que sí, si no, ya estaría, me dirían, pasa por haberes y ahí está.

A ver, rápidamente y repasando rápidamente, el sistema de nombres de dominio, o DNS, como lo conocemos por su sigla en inglés, es esa aplicación, que fue diseñada para traducir nombres a direcciones, nombres a identificadores únicos de Internet. Los identificadores únicos de dispositivos de Internet son las direcciones IP, que no son ni los únicos identificadores, ni tampoco son únicos, pero bueno, es la principal característica que debe tener un dispositivo que se conecte a Internet, es tener una

dirección, un identificador que lo diferencie de todo el resto de los dispositivos, y esa es la dirección IP. ¿Sí?

Y los nombres de dominio son como el equivalente a los nombres de personas en el directorio telefónico. ¿Sí? El DNS hace las veces de directorio telefónico para Internet. Traduce nombres de dominio a direcciones, ¿por qué? Porque son más fáciles de recordar que los números. Ya, simplemente por eso. Y otras montón de ventajas más, muchas de ellas técnicas y operativas que vinieron después con eso, pero bueno, la motivación principal de tener un DNS es esa, traducir nombres a direcciones.

Entonces, ahora vamos a hablar un poquito de cómo se hace que un sistema que originalmente traducía nombres que solo se podían escribir en ASCII, a algo que puede tener virtualmente cualquier tipo de lenguaje, digamos, codificado de una forma muy especial, que ahora Carlos me va a ayudar a tratar de explicar, digamos, ¿no? Porque en UTF-8 es como, puede llevar una semana a entenderlo. Bien. Y hablar un poquito también de cómo se lleva eso al... al correo electrónico.

Entonces, con la aceptación universal, entonces ahora va a ser posible tener, no, ahora no, hace unos cuantos años ya, es posible tener nombres de dominio y direcciones de correo electrónico en idiomas, cualquier idioma, ¿sí? Lo que hay que hacer es crear como el diccionario, digamos, ¿sí? La codificación para poder soportar eso. ¿Querés que lo... ahora o te corto el mombo?

Carlos Martínez: Dale. Bueno, el problema es el siguiente. Todos escucharon... No digo que... No digo que sepan lo que es. ¿Escucharon hablar de UTF-8 alguna vez? ¿Nadie? ¿Vos sí? ¿Vos también? Te estoy mirando. Bueno, el tema es así. ¿ASCII saben lo que es? ¿O les suena? ¿Les suena más seguramente? Bueno.

En definitiva, cualquier sistema de comunicación digital que se conecta, digamos, que se comunica con ceros y unos, y no me estoy refiriendo necesariamente a computadoras, de hecho el ASCII proviene del TELEX, el origen del ASCII viene del TELEX, en la década del 60, 70, 50, es una forma de representar caracteres en información digital. La primera versión de ASCII eran 7 bits, ni siquiera eran 8. ¿Por qué 7? Porque ni siquiera existía mucho el concepto de que 8 era un tamaño razonable para una computadora digital. Y en esa época cada bit, no cada byte, cada bit valía un montón de plata. Transmitir un bit apropiadamente a lo largo de una distancia larga valía un montón de... Costaba un montón de plata.

Entonces, este... Un organismo en Estados Unidos, creo que era el antecesor de la FCC de alguna forma, dijo, bueno, hace falta una forma estándar de representar caracteres para poderlos mandar por TELEX. Y se decidieron en 7 bits porque en 7 bits, en 128 combinaciones, dijeron, acá podemos codificar todo lo que nos hace falta para lograr una comunicación inteligible.

Nico dijo una cosa muy bien, que la gente asuma que ASCII es para representar inglés y no es así. ASCII es para representar el mínimo subconjunto, de inglés, que permite transmitir algo que más o menos se entienda. Si ustedes buscan en Google, por ejemplo, imágenes de telegramas viejos, en inglés incluso, van a ver que ni siquiera se transmitían todos los caracteres. Muchas veces se ahorraban las vocales. En las palabras donde se entendía igual se ahorraban las vocales. Cada letra costaba un montón de plata. Entonces, bueno, ese bit que... No es que se lo sacaran, nunca se les ocurrió ni siquiera ponerlo. Dijeron, ¿cuánto es lo mínimo, mínimo, mínimo que precisamos? 7. ¿Cuánto es lo mínimo, mínimo, mínimo que precisamos? Porque aparte de los caracteres, minúsculas, minúsculas, números, puntuación, iban determinadas secuencias de control. O sea, iba todo. En realidad había una tira de bits que se transmitía y ahí tenía que ir todo, incluso la información de control, donde empieza el mensaje, donde se acaba el mensaje, etc. Bien.

Con el advenimiento de las computadoras digitales, hubo un periodo de transición donde la gente no se ponía muy de acuerdo cuál era la agrupación de bits que era razonable. Hubo arquitecturas muy raras, de 13 bits, de 17, de 36, cosas por el estilo. Y en algún momento medio que la gente llegó a la conclusión de que 8 era un buen número. Entonces, ahí aparece el concepto de byte. Recién ahí aparece el concepto de byte.

Corren algunos años y la gente dice, bueno, el ASCII este que venimos usando, pero lo venimos usando porque es lo único que había, ¿no lo podríamos extender a 8 bits y usar ese bit que no estamos usando para algo? Bueno. Y ahí surge lo que seguramente ya, el ASCII que todos ustedes conocieron, que es un poco más rico que el ASCII original.

Ahora bien, siguen pasando años y se empieza a popularizar la computación personal, viene el Windows. Y en el Windows dice Microsoft, estaría bueno poder hacer esto multilinguaje, porque tenemos clientes en todas partes del mundo, pero no hay cómo representar esos caracteres. Y entonces inventaron un hack, literalmente un hack, con el cual venimos sufriendo hasta ahora en determinadas aplicaciones, que eran lo que llamaban los Windows Code Pages. ¿Sí?

Entonces, ellos lo que hacían, era mantenían los 7 bits del ASCII como si fueran ASCII. Pero dependiendo de, no me acuerdo qué valor, los siguientes 128, o sea los 128 que ganaste agregando un bit, lo usaban para representar caracteres en otros idiomas. Entonces, por ejemplo, en castellano y otros idiomas latinos, eran creo que el CP 1259, una cosa así. Pero si te ibas a Rusia, ya era otro. Y en China era otro, etcétera, etcétera. Y a su vez esos 128 que ganabas, no te daban para representar idiomas, por ejemplo con el chino, que tenés un montón de símbolos más. Claramente es un problema eso, y ahí vino también un esfuerzo de estandarización, que fue lo que derivó en algo que se llama Unicode, que está nombrado por ahí creo que en el slide anterior.

El Unicode es seguir extendiendo esa idea, representar caracteres, o ya cosas más genéricas como emojis por ejemplo, figuritas, eso tengo mi opinión muy personal, acerca de eso. No tengo la necesidad de eso. Pero bueno, seguir representando esas cosas como, ya no como un byte, porque no te alcanza con un byte, sino consecuencias de byte. Pero siempre manteniendo ese criterio de eficiencia que les nombraba antes.

Entonces, simplemente una aproximación simplista sería, ah bueno, si con un byte no me alcanzó, le pongo dos, o le pongo tres, o le pongo cuatro. El tema es que si yo cada carácter que voy a usar, va a usar cuatro bytes, efectivamente las cosas se me vuelven gigantescas de golpe. Puede ser más o menos un problema ahora eso, pero lo que pasa es que cuando se empezó a pensar en Unicode, efectivamente el almacenamiento todavía era una de las cosas caras, y el costo de transmisión también. El costo de transmisión, el costo del costo de transmisión se va a estar volviendo a volver un tema, que ahora les cuento después.

El tema es el siguiente, entonces la gente esta de Unicode dijo, bueno, ¿cuánto es el espacio que precisamos para representar todo? Es decir, todos los caracteres de todos los idiomas conocidos más, el emoji de carita sonriente, y el emoji del mate, y qué sé yo, y llegaron a un número que no sé cómo llegaron, pero es un número medio raro, es un millón 111 mil no sé cuánto, que en realidad en Hexa queda escrito 10 FFFF, ese es el máximo.

Entonces bueno, ahí ustedes se la pueden bajar, la base de datos de Unicode, en cual están todos los mapeos de esas cosas, leer algunas partes de eso es re interesante, porque cada carácter viene con lo que se llama su code point, que es el número que lo representa, una descripción en texto, por ejemplo la ñ dice latin n with tilde, esa es la descripción que tienen en inglés para la ñ, y así de todas, todas, todas, todas, todas, todas, ahí es cuando me enteré de eso, que hay caracteres

anchos y angostos en algunos scripts, y quieren decir cosas distintas, o sea el mismo carácter, ancho y angosto, quiere decir cosas distintas.

Entonces, sumado a eso, una vez que tenemos esos caracteres representados, aparece la necesidad de transmitirlos a través de un canal de comunicación. La aproximación simplista sería, cuántos bits necesito para representar ese número raro de un millón no sé cuánto, y más o menos te alcanza con, y va a estar bien, tenés que expresar 4 bytes. Como eso es súper ineficiente, se crea el concepto de lo que llaman las reglas de codificación, que es cómo representar ese número de 32 bits, de la forma más eficiente posible, en bytes, para transmitir y para almacenar.

Y ahí es donde aparecen las UTFs, hay varias UTFs. Unicode Transform Format, ahí está UTF, Unicode Transform Format. Entonces hay, la más común, que es la que se usa en DNS, en el correo electrónico, es UTF-8. ¿Sí? No es porque es, todo el mundo quiere que se ponga, porque el 8S tiene que ver con el byte, en realidad no tiene que ver con eso, es simplemente, de hecho es una codificación de largo variable.

En UTF-8, los caracteres ASCII, se representan, los que son, que ya existían en el ASCII original, se representan como su valor en ASCII. Y los otros, los raros, ahí se codifican, dependiendo de en qué punto están, en ese rango de 0 a 1.100.000, no sé cuánto, pueden ser, pueden ser codificaciones de 1, de 2, y de 3, no sé si hay de 4, capaz que haya de 4 también bytes. Y hay unos bits, en el medio, que se usan para decir, bueno, qué es lo que viene después, si es un carácter nuevo, o es un pedazo del que estoy armando. Entonces ese es el famoso UTF-8.

Hay UTF-7, que está deprecado, no se usa más, o no debería usarse más, siempre uno se lleva unas sorpresas por ahí. Las cosas, cuando uno depreca cosas, uno puede estar seguro, que dentro de 35 años, vas a ir a abrir algo, y va a estar codificado en eso. Y el que sí se usa, en algunas aplicaciones, es una cosa que se llama UTF-32, que es una codificación muy simplista, que es eso, es decir, agarrar cada carácter, y luego, la ventaja de eso, los que alguna vez lucharon con programar, saben que procesar cosas de largo fijo, es en general mucho más fácil que procesar cosas de largo variable. Entonces, en determinadas aplicaciones, se prefiere el UTF-32.

Así que bueno, este es el primer, como decirle, el primer secreto oscuro de todo esto que les vengo a contar. Ahí está.

Nicolás Antoniello: Y para continuar un poco, con lo que venía diciendo Carlos, como me gustó el ejemplo de los emojis, es un ejemplo muy bueno para los que conocen más, están más acostumbrados a... Ustedes saben que en muchas aplicaciones, digamos, para poner una carita riéndose, pueden escribir, no sé, dos puntos, LOL, y eso quiere decir laughing out loud, o riéndose mucho, digamos, y dos puntos de nuevo. O sea, lo ponen entre dos puntos, y lo que está en el medio, la aplicación lo interpreta como, ah, esto lo tengo que codificar, es una codificación, no es que quiera transmitir dos puntos, LOL, dos puntos, que no significa nada, sino que es una carita sonriendo. Entonces lo sustituye por una carita sonriendo.

En el DNS es igual, digamos. A través del UTF-8, y de una codificación que para el DNS, para el tema de aceptación universal, se llama Puny Code, se escribe utilizando caracteres ASCII, algo que después las aplicaciones y los sistemas lo interpretan como lo que debe ser, que es un carácter internacionalizado, quiere decir, internacionalizado quiere decir en cualquier cosa que no sea ASCII, digamos, ¿sí?

Este, la verdad que estoy pensando es un poco, este, como, no importa, internacionalizado cuando no es ASCII, es como cuando decimos, voy al interior, y de repente voy a Piriápolis, Piriápolis no está más, no hay nada más afuera que Piriápolis, porque está sobre la costa, pero para nosotros es el interior, ¿no? Bueno, esto es igual. Internacionalizado quiere decir que no es ASCII, está bien, ponele.

Bueno, entonces, y así se codifica, en el DNS y así se codifica en el, en el, en el correo electrónico, y es un poquito, un poco también lo que, lo que hace el formato MIME ese, que, que permite hasta agarrar una imagen y convertirla en texto y transmitirla como texto y que después del otro lado se reconstruya y sea una imagen, digamos, ¿no? Entonces, para mandar una imagencita así, genero 57 hojas de, de MIME, de texto, y después eso, otra vez, este, en forma absolutamente ineficiente, a menos que lo comprima, lo, lo vuelvo a traducir a la imagen.

Entonces, yendo un poco al, al correo electrónico y entrando más en el tema del correo electrónico, a modo de repaso, este, estos son como tres grandes partes de un correo electrónico, el, lo que sería el sobre, este, el encabezado del, del, del mensaje y el cuerpo del mensaje, digamos. El, el correo electrónico, cuando, cuando lo hicieron, era formato electrónico, el correo postal tradicional, ¿no? Entonces, este, el, el sobre virtual, en el caso del correo electrónico, tiene afuera un, un remitente y un destinatario, digamos. Esos, esos son, esos son parte de los encabezados del mensaje, digamos, y lo usa el sistema de envío y de, y de, ¿cómo se dice? Delivery, sería como, sí, como envío. El sistema de envío del correo

electrónico usa esos encabezados y esas, esa información para hacer que el, el correo electrónico ¿sí? Eh, por ejemplo, de un cliente de correo electrónico donde yo redacto un correo electrónico a la casilla de correo del destinatario, pasando por un montón de servidores y dispositivos en el medio.

Y el cuerpo del mensaje es lo que va dentro del sobre, ¿sí? Ya les digo que aceptación universal no es meterse en el cuerpo del mensaje. Si el cuerpo del mensaje está escrito en el idioma que esté escrito, al, al, al, al proyecto de aceptación universal no le interesa porque eso es lo que va dentro del sobre. Así como no le interesa entre comillas al cartero lo que va dentro del sobre, ¿sí?

En honor a la verdad, eso fue un problema en una época. Pero eso se resolvió bastante antes cuando, en general, casi todos los sistemas operativos más o menos en una ventana de tiempo fue alrededor de fines de los 90, adoptaron el UTF-8. Todos los sistemas operativos se volvieron UTF-8. Entonces, lo que va dentro del mail o de un Word o del archivo de texto, incluso, dejó de ser un problema. Siguió siendo un problema durante mucho tiempo los nombres de archivo, por ejemplo, y bueno, y esto que sigue siendo un problema hasta ahora.

Lo que no quita que no sea un problema por temas de seguridad, etcétera, etcétera. Pero bueno, eso es lo mismo. Yo puedo mandar una bomba dentro de un sobre y puedo mandar un archivo comprimido dentro de un correo electrónico que sea una bomba digital, digamos, ¿no? Algún tipo de ataque, virus troyano. Hoy en día tenés una familia interminable de daños que se pueden hacer con un correo electrónico, ¿no? Pero bueno, a los efectos de aceptación universal no nos vamos a concentrar en lo que está adentro, sino lo vamos a concentrar en lo que está fuera del sobre. ¿Sí?

Bien. Estos son definiciones que cuando crucen la puerta si se las olvidan su vida va a ser mejor. Son como los distintos dispositivos y agentes que participan en la cadena de envío y recepción y transmisión de un correo electrónico, ¿sí? Entonces está el MUA ese que es el, no es otra cosa que una sigla muy complicada para describir al cliente de correo electrónico, que puede ser cualquier cliente de correo electrónico, no sé, Outlook, ¿cómo se llama el de Apple? No me acuerdo ya, pero bueno, hay un montón, el que quieran ustedes.

En la época, acá estoy mirando a Martín que trabajó, en otra vida trabajamos juntos en algún lado, que usábamos Paint, ¿te acordás de Paint? Un cliente de correo electrónico en modo texto que era una cosa solo para nerds, digamos. Sí, sí, se hacía todo con las teclas así. ¿Eh? No era el peor. Y no era el peor, había peores,

había peores, había peores. Tenías alto de... Sí, sí, sí, hacías cosas mucho más rápido hoy en día y para algunos, este, ojalá hubiera seguido esa, esa... Para los nostálgicos del Paint sigue existiendo el MUT. Bueno, lo vamos a usar hoy de tarde para una demostración, así que está.

Este, el MSA, que eso ya está del lado del servidor, digamos, este, el correo, en realidad, en el, en el, es parte del sistema que envía el correo electrónico que uno genera en el MUA, digamos. El MSA es el que inicia el envío, el primero que envía el correo electrónico generado, digamos. Después, MTAs son todos los, vamos a hacerlo, a hablarlo mal y rápido, digamos, pero para que quede claro, son todos los sistemas por los que va pasando el correo electrónico. Todos los servidores intermedios por los que va pasando un correo electrónico cuando viaja del origen a un destino.

O sea, van a ver, típicamente, va a haber un MUA, un MSA, varios MTAs y un MDA al final, que es el que recibe el correo electrónico y el MDAs es el programa que agarra el correo electrónico que viene y lo pone, lo deja en la casilla de correo, lo deposita en la casilla de correo del usuario destinatario, digamos. ¿Sí?

Bien. ¿Por qué hablamos de todo eso, además, no? Ahí apareció el, no era IEA, sino que es EAI. Bueno. Tener soporte UTF, cosas que tienen que pasar, que ya explicó un poco Carlos, hay que tener el soporte UTF-8 para poder generar estas codificaciones en otros lenguajes que no sean ASCII, tanto para el nombre del buzón como para el nombre de dominio.

soporte de dominios universalizados, mi, todo mi sistema de DNS tiene que soportar dominios internacionalizados y esto no es solo que mi servidor DNS donde pongo la información que se llaman servidores autoritativos tengan soporte de esto, sino que los servidores recursivos también tienen que tenerlo, tengo un montón de, todos los servidores que participen de la resolución del nombre de dominio tienen que soportar eso.

Y lo mismo, el, si yo quiero tener un nombre de buzón internacionalizado, aunque no tenga un dominio internacionalizado, por ejemplo, nicolascontildela arroba internet.com.uy o lo que sea, voy a necesitar que todos los servidores que participan del correo electrónico, o sea, toda esta familia de servidores de la que hablábamos, todos ellos, por donde pase ese correo electrónico, tienen que soportarlo. Si el correo electrónico llega a un servidor intermedio, una MTA intermedia, por ejemplo, que no soporta eso, ahí murió. Hasta ahí llegó. ¿Sí?

Y lo que pasa después de eso es una ruleta. Hay algunos sistemas, esto es un poco opinión personal, que están bien configurados, que envían una respuesta hacia atrás para que el origen sepa la razón por la cual no puedo pasar de ahí. Básicamente, el que lo rechace va a decir, yo no soporto universalización de casilla de correo o de nombre de dominio o ninguna de las dos. Hasta acá llegó el correo, pero te aviso que fue por eso que lo voy a descartar. Si descarte, chau. Y ese correo no va a llegar nunca a destino, digamos. No importa cuántas veces lo envíe. Entonces, toda la cadena tiene que soportar aceptación universal. Eso es uno de los obstáculos que hay para desarrollarlo. No depende solo de mí, depende de todo lo que está en el medio.

Y bueno, hay algunas cosas que no es aceptación universal. Digamos, UTF-8 ya existía muchísimo antes y se utilizó. Es una herramienta más que se utilizó también para este tema. El contenido del asunto del correo electrónico no es de interés para enviar el correo electrónico. De hecho, no le interesa, solamente le interesa al quien lo envía y al que lo recibe. Comentarios de dirección. Esto no vamos a entrar en detalle, pero bueno, la dirección se repite dos veces. Va fuera del sobre para que la use el cartero y va dentro del sobre de nuevo. Que la que va dentro del sobre no importa. La que va dentro del sobre incluso hasta podría mentir en lo que va dentro del sobre, pero al cartero lo que le importa es lo que está fuera del sobre y aceptación universal también. Y obviamente el cuerpo de mensajes tampoco, porque va dentro del sobre y eso lo puedo escribir en el idioma que quiera.

Y bueno, para la universalización de todo eso que no es exactamente aceptación universal, ya existe MIME que combinado con el UTF-8 te permite representar casi cualquier cosa. Por ejemplo, en el cuerpo un mensaje del correo electrónico, incluso una imagen, un documento de algún procesador de texto, una presentación PowerPoint, un PDF, cualquier cosa.

Bien. Entonces, yendo un poco más al detalle y a la configuración, porque ya estamos casi que sobre la hora, pero vamos a cinco minutos más. Algunos algunos requisitos a esto la presentación se la dejamos para que los que son más técnicos, los más curiosos lo tengan lo tengan lo tengan detallado esto es como el algunas cuestiones que hay que tener en cuenta cuando yo estoy configurando aceptación universal o cuando quiero dar soporte de aceptación universal.

Bueno, ahí es como una recetita hay que normalizar la cadena Unicode antes de procesarla en UTF-8 utilizar el formato correcto para para para IDN y bueno, hay varios formatos de muchos de estos estándares que se utilizaron al comienzo pero ya no se no se recomienda más utilización sino que se recomienda utilizar siempre

la última versión digamos de todas estas cosas si por ejemplo el el IDN a muchos muchas aplicaciones están programadas utilizando este esta versión la versión 2003 esta versión ya no se recomienda utilizar más entonces ahora los que ya tienen esto programado tendrán que sacar algún parche con esta versión y los que están empezando a programar No usen esta, usen aquella versión de IDNA.

Carlos Martínez: ¿Contaste qué es IDNA? No, contale. Se han dado cuenta que este asunto del Unicode te deja hacer muchas cosas y representar muchas cosas. En realidad te deja hacer tantas cosas y representar tantas cosas que se vuelve un peligro, en cierto sentido. Entonces, para los efectos de ser usado en Internet, para comunicaciones en Internet, no toda la variante, la enormidad de variante de cosas que te deja representar Unicode están permitidas. Entonces hay una serie de reglas que hacen a qué cosas sí son válidas y qué cosas no son válidas para mandar por Internet. Que sí serían válidas dentro del cuerpo del mensaje, un poco lo que estaba diciendo Nico, de repente.

La más obvia, por ejemplo, es que en los dominios de alto nivel, los top level no se permiten emojis. O sea, no va a haber nunca, o bueno, mientras nosotros estemos vivos, no va a haber, sobre mi cadáver, va a haber un dominio de alto nivel que sea punto mate, digamos. Punto mate, dibujito de mate, quiero decir. Ese es un ejemplo chistoso, pero de esas cosas hay muchas. Porque hay todo un tema de seguridad vinculado a esto que capaz que lo podemos conversar, que es el de los ataques homógrafos. Que es tremendo.

O sea, la cantidad de cosas que se pueden hacer y de engaños que se pueden hacer en un mundo donde hay gente que te llama por teléfono y te pide plata y la gente se la da. O sea, imagínense las cosas que se pueden hacer dibujando con caracteres raros. Es increíble. Entonces, bueno, hay una serie de reglas que dictan un poco qué cosas son válidas de representar en direcciones de correo, nombres de dominio, sobre todo nombres de dominio, que tienen que ver con esas reglas de IDNA. Además de algunas reglas de codificación adicionales y eso.

Nicolás Antoniello: Bien. Me fui hasta el final de la presentación porque el resto de lo que está en el medio de la presentación lo vamos a ir viendo en la tarde a medida que vayamos haciendo el laboratorio demo. Vamos a hacer una especie de demo con Carlos Guiada en la tarde, después del almuerzo. Y ahí vamos a ir pasando la presentación.

Solo quería dejarles esto a modo de chequeo si tienen un rato ahora o cuando quieran. Esta es una página que está disponible en el sitio de... del Universal

Acceptance Steering Group, digamos, que les puede servir para chequear si su servidor de correo electrónico admite la internacionalización de correo electrónico. O sea, si soporta básicamente aceptación universal. ¿Sí? Y acceden ahí. Y básicamente van a generar un correo, lo van a mandar y el sistema va a ver si obtiene respuesta y va a ver después qué nivel de respuesta obtiene.

Porque dentro del tema de... de la aceptación universal y el soporte de los sistemas... Perdón, de correo... Este es el problema de comer... Ah, maicena antes de... El correo electrónico tiene varios niveles de soporte. Y no quiere decir que haya que cumplir todos. O sea, que haya que tener soporte absoluto y completo para poder transmitir un correo. Por ejemplo, si es un MTA, que es un agente intermedio, que ni es el que genera el correo ni es el destinatario y no soporta crear una casilla de correo... universalizada... Bueno, no importa, porque el MTA en realidad no es donde se crea la casilla ni donde se maneja eso, sino que el MTA tiene que recibir un correo y pasárselo. Sería como una especie de agencia de correo intermedia.

Pero sí tiene que tener soporte a eso. Tiene que entender que es un carácter internacionalizado en el nombre de una casilla de correo o en un dominio para poder hacer el reenvío del correo electrónico a su destino. Entonces hay diferentes niveles. Entonces, un poco con esta herramienta se puede chequear si soporto y a qué nivel lo soporto. Y eso tiene varios tipos. Tipo 1, tipo... En general es tipo 1 soporte tipo 1 y soporte tipo 2, digamos. Tipo 2 es el que soporta todo. Tipo 1 es un soporte parcial que se puede utilizar, por ejemplo, en un MTA, pero no se puede utilizar ni para un agente de cliente de correo electrónico ni para un destinatario de correo electrónico, digamos.

Y bueno, esto ya lo hablamos un poco al principio. Más allá del correo electrónico, la idea es que... Como... Invitarlos a... Invitarnos a todos a eso, a revisar no solo la parte del software en sí, sino a nivel de aplicaciones, a nivel de interfaces web de usuarios, si estamos o no estamos soportando la aceptación universal.

De hecho, no basta solo con el software, porque si nos olvidamos de todos los formularios, por ejemplo, para crear una casilla de correo, si la interfaz de... Hoy en día hay, por ejemplo, todos los servidores de DNS que se utilizan hoy en día, autoritativos y recursivos, en todas sus versiones, en todos sus sabores, digamos, de cualquier fabricante, en las versiones desde hace, no sé, cinco años a la fecha, soportan universalización de los nombres de dominio.

Se sorprenderían de la cantidad súper reducida de registradores, agentes que comercializan nombres de dominio, que soportan la aceptación universal, que soportan la universalización. Son muy, muy pocos. Todos sus sistemas de DNS ya lo soportan. Lo que no lo soporta es la interfaz web, ni los sistemas de facturación, ni los sistemas de procesamiento. Entonces también hay que hacer una inversión en eso, que creo que es la más grande, porque el resto es actualizar el software y ya está.

Por supuesto que, como todo, decirles una cosa, hacerles otra, hay que formar a la gente, hay que capacitar a los grupos de operaciones, etc. Pero un poco la idea de estas reuniones también es eso, es por lo menos dar el puntapié inicial para que ustedes empiecen a explorar esto. Bueno, hasta acá por ahora.

Moderador: Bueno, muchas gracias Carlos y Nicolás. No les dije, pero Carlos y Nicolás obviamente son miembros del capítulo, miembros fundadores, y Carlos fue el presidente de ISOC-Uruguay también, cosa que me olvidé de comentar, pero bueno, todos los conocemos y la verdad que les dije que eran un buen dúo haciendo presentaciones. Bueno, ahora siguiendo con este tema, hablando del tema de los emojis que comentaste, Carlos, en la GNSO en ICANN hay varios grupos de discusión acerca de la inclusión de los emojis.

Carlos Martínez: Tentador, o sea, abre un montón de posibilidades de negocio. Muchas. No muy genuinos, pero bueno, negocios al fin. Pero sí, seguro que es un tema de discusión, pero también genera todo un mundo de problemas gigantescos.

Moderador: Sí, estaba justamente los a favor, los en contra, varias discusiones al respecto muy interesantes.